SEND EVENT

Note:
This statement is only available under Windows.

Operand	Possible Structure					Possible Formats												Referencing Permitted	Dynamic Definition
Operand1	C	S				A												yes	no
Operand2		S							I									yes	no
Operand3	C	S	A			A	N	P	I	F	B	D	T	L	C	G	O	yes	no

Function

You use this statement to trigger a user-defined event within a Natural application.

Operands

Operand1 is the name of the event to be sent.

Operand2 is the identifier of the dialog receiving the user event. *Operand2* must be defined with format/length I4.

AD=

If operand3 is a variable, you can mark it in one of the following ways:

AD=O	Non-modifiable, see Session Parameter AD=O.
AD=M	Modifiable, see Session Parameter AD=M. This is the default setting.
AD=A	Input only, see Session Parameter AD=A.

Operand3 cannot be explicitly specified if operand3 is a constant. AD=O always applies to constants.

Passing Parameters to the Dialog

It is possible to pass parameters to the dialog.

As *operand3* you specify the parameter(s) to be passed to the dialog.

With the *PARAMETERS-clause*, parameters may be passed selectively.

nX

With the notation *n*X you can specify that the next *n* parameters are to be skipped (for example, 1X to skip the next parameter, or 3X to skip the next three parameters); this means that for the next *n* parameters no values are passed to the dialog.
A parameter that is to be skipped must be defined with the keyword OPTIONAL in the dialog's DEFINE DATA PARAMETER statement. OPTIONAL means that a value can - but need not - be passed from the invoking object to such a parameter.

PARAMETERS-clause

```
PARAMETERS {parameter-name = operand3}... END-PARAMETERS
```

Note:
You can only use the PARAMETERS-clause if the specified target dialog (dialog-name) is cataloged.

Dialog-name is the name of the dialog receiving the user event.

Note:
If the value of a parameter marked with AD=O and passed "by reference" is changed in a dialog, this will lead to a runtime error.

Further Information and Examples

See the section Event-Driven Programming Techniques in the Natural Programming Guide.